**NLP COURSE PROJECT**

**Hate Speech Detection in Tweets**



**Submitted By:**  
 Aaditya Rathod IIT2020209

<https://github.com/aadityarathod7/NLP_project/tree/main>

(GitHub Link)

1. ***Abstract*—**Twitter's central goal is to enable everybody to make and share thoughts and data, and to communicate their suppositions and convictions without boundaries. Twitter's job is to serve the public discussion, which requires portrayal of a different scope of points of view. Yet, it does not advance viciousness against or straightforwardly assault or undermine others based on race, nationality, public cause, rank, sexual direction, age, inability, or genuine illness. Hate Speech can hurt a person or a community. So, it is not appropriate to use hate speech. Now, due to increase in social media usage, hate speech is very commonly used on these platforms. So, it is not possible to identify hate speeches manually. So, it is essnetial to develop an automated hate speech detection model and this resaech work shows different approaches of Natural Language Processing for classification of Hate Speech through Machine Learning Algorithms.
2. **INTRODUCTION**

After the advent of social media and its increasing number of users, there is less and less control over the content posted by them. Hate and abusive speech being one of these. People are now able to express or show any type of emotion online which was once difficult when talking to a person face to face. You can see hate content being posted daily in the form of comments, images, videos etc. Automated hate speech detection plays a huge role when dealing with a huge number of users. Here we will be using machine learning classification models to classify hate comments and tweets.

1. **LITERATURE SURVEY**

The research papers chosen for study use various classification models.The dataset used in these research papers is from twitter. The models used in these research papers were tested on these tweets and the classification was done accordingly.The criteria for selection of the research papers was the classification models used and their accuracy so that the knowledge gained through them could be used in our study.The models used in the report have been used in these base papers. The paper published by Joni Salminen1,2\* , Maximilian Hopf3 studies the classifier results on multiple platforms thus supporting the goal of developing more universal online hate classifiers for multiple social media platforms which uses Logistic regression (LR), Naïve Bayes (NB), Support‑vector machines (SVM), XGBoost, Bag of words, TF-IDF, BERT models for classification.According to their study XGBoost gives the best results on all the features whereas the other models (LR,SVM and NB)performed bad.

The paper published by Thomas Davidson,1 Dana Warmsley uses TF-IDF,logistic regression with L1 regularization, naive Bayes and Random Forests .This paper uses various linear classification models for multiclass distinction. After using grid search, models like SVM and Logistic regression have performed better.

1. **METHODOLOGY**
   1. DATASET

Tweets were gathered by Davidson et al.[1] who then voted on the class to give each tweet.24,783 tweets that were categorized as hate speech, offensive language, or neither make up the dataset.Hate speech is represented by class 0, offensive speech is represented by class 1, and neither is represented by class 2.

|  |  |
| --- | --- |
| **Class** | **No. of Instances** |
| 0 | 1430 |
| 1 | 19190 |
| 2 | 4163 |
| **Total** | **24783** |

We have followed a step by step approach to prepare our results. The first step was to choose the dataset to test our model.Next we have done preprocessing using stop words,stemming,tokenization etc.

Before testing our model on the dataset feature extraction was done to get numerical data using TF-IDF vectorization.Next we finally tested our data on 3 models NB, Random Forest, XGBoost and calculated metrics like F1 Score,recall,precision.

* 1. PREPROCESSING

Tokenizing:Breaking sentences and paragraphs into smaller chunks to better convey the idea and develop a vocabulary.

Stemming: The process of taking care of all prefixes and suffixes in order to provide a single root word the meaning of all related terms.

Stop Words: Eliminating offensive words.

Case Folding: All words change



Fig1: Word Cloud for offensive class

* 1. FEATURE EXTRACTION

The technique of turning raw data into numerical features that can be handled while keeping the information in the original data set is known as feature extraction. Compared to using machine learning on the raw data directly, it produces better outcomes. We must translate the text into numbers because the study deals with text.

To determine the frequency of each word appearing in a document, use the Term Frequency-Inverse Document Frequency (TF-IDF) method.

**Term Frequency**: Number of times the term "t" appears in the document / (number of terms in the document)

**Inverse Document Frequency (IDF)** is defined as log(N/n), where N is the number of documents and n is the number of documents where a term (t) has appeared.

**TF-IDF value of a term** = TF x IDF

1. **METRICS**

The***accuracy*** of the model's output is determined using the accuracy metric.

*True positive:* The positive class was successfully predicted by the model.

*True negative:* The negative class was accurately predicted by the model.

Positive class suggested by the model was a *false positive.*

*False negative:* The negative class was accurately predicted by the model.

*Precision:* The ratio of true positives to the total of both true and false positives is known as precision.

*Recall:* The proportion of true positives to the total of true positives and false negatives is known as recall.

*F1 Score:* This represents the harmonic mean of recall and precision.

1. **ML MODELS**

**Naive Bayes**:

It uses the Bayesian formula for predicting the class of the datapoint. It uses the prior probability. We can calculate the probability of event p given q has occurred from the probability of occurrence of event p has occurred.We assume feature independence here.

**XG Boost :**

XGBoost is an optimized distributed gradient boosting library designed for efficient and scalable training of machine learning models. It is an ensemble learning method that combines the predictions of multiple weak models to produce a stronger prediction. In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost.

**Random Forest**:

Yet again a classification algorithm,random forest exploits all the classification models to pick the one that gives best results.It works on the principle ‘Weak learners combined together will be strong’.We usually pick a huge dataset and make multiple datasets from it using sampling with replacement.Each of these small datasets are passed through different prediction or classification models like logistic regression,svm etc.Then we take the majority voting i.e. we pick the class that most models favor on these small datasets. The most favored class will be chosen as the final prediction of the random forest model.

1. **RESULTS**

3 Categories (Hate Speech, Offensive Language, Neither):

* + XGBoost: 96%
  + Random Forest: 97%
  + Naive Bayes: 86%

1. **References**

[1]<https://github.com/t-davidson/hate-speech-and-offensive-language>

[2]Salminen, J., Hopf, M., Chowdhury, S. A., Jung, S. G., Almerekhi, H., & Jansen, B. J. (2020). Developing an online hate classifier for multiple social media platforms. *Human-centric Computing and Information Sciences*, *10*(1), 1-34.

[3] Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. *Proceedings of the International AAAI Conference on Web and Social Media*, *11*(1), 512-515

[4] A. Tiwari and A. Agrawal, "Comparative Analysis of Different Machine Learning Methods for Hate Speech Recognition in Twitter Text Data," 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT), 2022, pp. 1016-1020, doi: 10.1109/ICICICT54557.2022.9917752.

[5]<https://www.capitalone.com/tech/machine-learning/understanding-tf-idf/>